

# 面向多源城市出行数据的可视化查询模型

汪飞, 陈为, 张繁, 鲍虎军

**摘要:** 多源城市出行数据为理解城市现象和享受城市生活提供了重要的基础。利用统计物理学和数据挖掘技术, 人们在多源城市出行数据的研究上取得了很大进展, 但是基于可视分析方法的研究甚少。本文提出了一种可视化查询模型旨在解决使用可视分析方法研究多源城市出行数据时所面临的挑战。查询模型从数据抽象、数据组织和管理、查询交互界面以及可视化分析等方面进行探讨, 介绍了一系列方法和技术。最后实现了查询模型的实例系统并分析了典型案例。

**关键词:** 多源城市出行数据, 可视化查询, 查询模型, 可视分析

## 1. 引言

世界上大约有一半的人生活在城市, 每天产生 **PB** 级出行数据, 如公共交通记录、车流量监控记录、出租车记录、智能手机记录、社交网络签到记录等。出行数据不仅成功应用于城市计算的基本问题, 如城市规划[1]、交通状况分析[2]、空气质量分析[3]、灾难分析[4]等, 而且在城市内人群运动规律上得出重要结论[5, 6]。

出行数据具有时空特性, 而且体量大、维度高, 出行数据更重要的特点表现在数据类型更多样而且数据粒度不均匀。例如, 同一手机用户相邻的通话记录有时相隔几分钟, 有时相隔数小时甚至几天。受出行数据特点的限制, 已有的分析方法往往限于某种特定类型的数据, 而且分析目标单一、分析周期较长。然而文献[3,7]表明不同来源的出行数据往往能够相互弥补单一数据源的偏差和数据缺失, 更全面地体现城市人群的行为模式。

利用数据挖掘技术人们在多源城市出行数据的研究上取得了很大进展[3,7,10], 但是可视分析方法应用于多源出行数据的研究还非常少见。可视分析方法提供图形化交互界面, 允许用户探索式分析大规模异构数据, 以相同的编码方式表示不同来源的数据不会产生歧义。面对多源出行数据, 可视化分析至少需要解决两方面的困难: ① 如何组织多源异构数据使得交互探索具有实时性; ② 如何可视化表达多源数据呈现的出行模式并与数据源无关。

本文提出了从数据组织、存储到索引、查询, 最后到交互界面、可视化和分析的统一可视化查询模型。模型涉及数据管理、数据分析和可视化等多方面技术。首先分析出行数据的基本结构, 抽象出一般化的出行数据表达形式; 在此基础上, 数据按时间、空间和出行对象分类并关联存储; 为提高数据查询的效率, 构建了基于哈希的双向链式时空索引; 此外本文还总结了出行数据的查询模式; 为便于数据探索, 本文探讨了设计可视化查询交互界面和可视化分析的主要技术。本文实现了统一可视化查询模型的实例系统, 利用此系统可以发现一些重要的出行模式。

## 2. 相关工作

城市出行数据分析方法主要有三类: 基于统计的数据分析方法、数据挖掘分析方法、可视分析方法。统计方法是利用统计学模型对人类行为进行定量分析, 力图发现新的统计规律。Gonzalez 等[8]发表在《自然》上的论文研究了 100 万手机用户 6 个月的通话记录, 发现人类的行为并非无规律的, 每个人都重复地回到一些高频率出现的位置。Song 等[9]继续研究发现手机用户位置的可预测性达到 93%。数据挖掘技术则对城市内多种数据进行整合, 运用相关技术分析和解决城市中存在的各类问题, 如城市规划[1]、空气质量预测[3, 10]、灾难分析[4]等。可视分析方法通过交互和图形元素表示数据, 分析人员可以自由探索数据, 发现数

据中的模式和规律。文献[2]分析了出租车 GPS 数据，发现城市中交通拥堵路段并推断形成拥堵的原因。文献[11]针对出租车 GPS 数据设计了可视化查询和分析工具，可以任意选择出租车上下车位置分析城市不同时间段的出行模式。尽管前两类方法应用非常广泛，但是二者都是单任务驱动的。也就是说，通常一个分析任务只有一个分析目标，分析目标改变则需要重新设定整个分析过程。而可视分析方法中分析任务和目标耦合性较低，特别有利于分析人员发现分析目标，而这往往是分析任务中最难的。

城市出行数据的存储和管理是数据分析的基础，主要采用时空数据管理技术。时空点数据管理和轨迹数据管理是两类基本方法。基于多维树索引是点数据管理的主要实现技术。3D R-Tree [12]、STR-Trees [13]、HR-Tree [14]是 R 树的扩展根据轨迹相似度建立最小包围盒(MBR)，并且单独增加 R 树管理时间维度。对于一些特殊的应用，研究人员进行了特别的优化。Nanocube[15]是针对可视化中聚集数据在空间使用四叉树，时间上则用求和区域表(Summed Area Table)压缩时间维度数据。轨迹数据管理受数据点的时序性约束，存储上需要具有连续性。SETI[16]和 TrajStore[17]空间上均匀或四叉树划分，而相同空间区域中的轨迹连续存储在相同或相邻物理页面。

### 3. 可视化查询

可视化查询是在交互图形界面上通过一系列选取操作执行查询，查询结果以图形化的形式展现。动态查询是可视化查询的早期形态，由用户通过图形化的选取工具直接操纵数据，动态获得查询结果[18]。随着可视分析的发展，可视化查询并不局限于直接操纵数据，而允许数据经过有效的组织和管理。可视化查询的优势在于：① 让用户直观地传递查询意图，避免了输入冗长的查询语句；② 快速获得图形化的查询结果，便于理解和指导下一步的查询操作。

可视化查询不仅是方便查询，更主要的目标是为了更好地分析数据。总体上，可视化查询包含三个部分：图形交互界面、数据查询模块、可视化分析模块。理论上，任何数据都可以通过可视化查询进行分析，但是数据规模和数据复杂性往往限制了可视化查询的应用。可视化查询不仅要求交互选取简便，而且需要实时的查询数据，更重要的是对查询结果进行分析并可视化地呈现。可视化查询涉及到数据组织和管理、数据挖掘、数据可视化和交互等多方面的知识和技术，只有三方面技术的有机合成才能构成有效的可视化查询系统。

多源出行数据对可视化查询提出了更大的挑战，一方面需要组织和管理复杂的多源异构数据，另一方面需要对多源数据进行一致化的分析和可视化。本文寻找多源出行数据的共性特征，抽象出统一的出行数据模型。在此模型基础上构建存储、查询模型，并执行一系列的交互式分析和可视化。

#### 4. 可视化查询模型总体结构

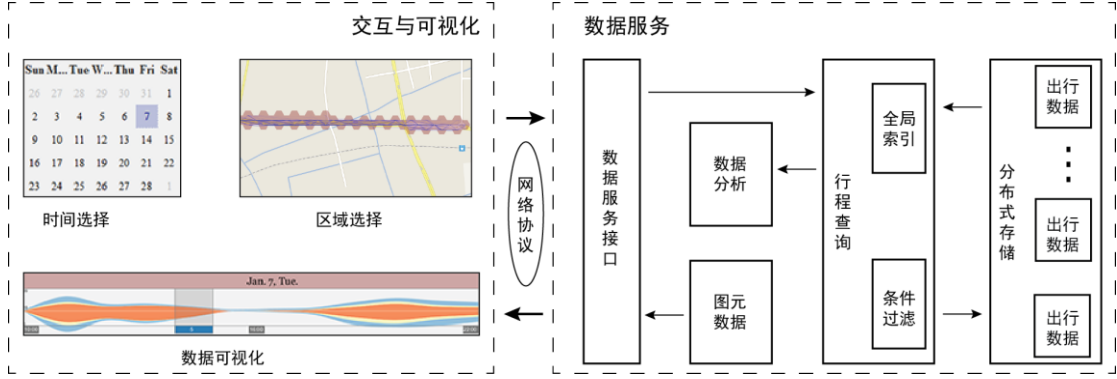


图 1 可视化查询模型总体结构

可视化查询模型总体可分为两部分，其中交互与可视化模块负责传递用户意图并且反馈分析结果供其理解和决策，而数据服务模块主要是获取查询数据并返回可以理解的数据图元。受数据规模和处理能力的限制，数据服务模块通常部署在性能高、容量大的分布式系统中。交互与可视化模块以图形化界面呈现给用户，可以同时存在于多个终端。两部分之间经网络协议传递少量数据，降低数据规模和复杂性对用户分析数据的影响。

交互与可视化模块为用户提供交互界面，包括时间选择、地图区域选择以及分析结果的可视化。分析结果的可视化是多维数据的视觉呈现，但需要包括两个基本的时间和空间维度。交互查询是迭代式的探索过程，在交互过程中用户根据可视化视图决定下一步的查询动作。

数据服务模块中包含数据存储、轨迹查询、数据分析和可视映射四项功能。出行数据按时间和空间分别存储于分布式系统中不同数据节点上。数据服务接口接收到用户终端发送的查询请求后，将请求中的查询条件传递至行程查询。后者将根据查询条件从全局索引中寻找适当的行程轨迹，并计算轨迹与查询空间约束的拓扑关系。数据分析主要实现分类、聚类、统计等计算，之后将计算结果传递给可视映射功能，将其转换为特定的图元数据。最后再次通过数据服务接口返回至交互与可视化模块。

#### 5. 数据抽象和数据存储

##### 数据抽象

尽管记录出行数据的设备各有不同，但是数据中一般都会包含出行的对象（人或车）、运动对象的位置、记录的时间以及一些其他属性，如车辆的状态、速度或进出站标识等。如果仅考虑这些数据的共性，那么多源出行数据模式可以统一定义为四元组的集合，即

$$\mathcal{M} = \langle O, L, \Gamma, \Omega \rangle$$

其中， $O$ 为出行对象集合， $L$ 为地理位置， $\Gamma$ 表示记录时间， $\Omega$ 为其他各类属性集合。

给定一个时间段，任意对象必将产生按照数据模式 $\mathcal{M}$ 的出行数据集合，称为行程。一般地，行程可以定义为

$$\gamma = \{(o_i, l_1, t_1, \omega_1), (o_i, l_2, t_2, \omega_2), \dots, (o_i, l_n, t_n, \omega_n) \mid i, n \in \mathbb{N}\}$$

而行程中每个记录则称为行程点，每个行程由时间连续的行程点集合组成。实际行程中，出行对象可以处于运动状态也可能处于停留状态，尽管本文主要研究前者但数据模型并不影响同时对上述两种状态的研究。处于运动状态的对象，在行程上通常表现为相邻行程点位置不同，即

$$\gamma_{i \cdot l_k} \neq \gamma_{i \cdot l_{k+1}}$$

受数据记录方式的影响，不同来源的行程在时间具有不同的数据粒度，有些时间间隔比

较均匀，而更多的时间间隔差异较大。比如，浮动车数据时间均匀间隔，通常几十秒到几十分钟记录一次；手机通信数据则时间跨度大，有的行程点间隔几十秒而有的则间隔数小时。另一方面，受出行交通工具和时间不均衡的影响，行程空间上跨度也极为不均衡。连续的行程点有的相隔几米，而有的则达到数公里。数据时空分布不均衡直接导致行程存储上的困难。

## 5.2 数据存储

为保证行程访问的效率，数据需要进行合理的划分。划分的基本原则是：相关性较强的数据物理存储上相邻[18]。从行程的数据特征来看，可以有如下四种划分方法：

- 1) 按时间分段存储，即按一定的时间段（小时、天等）将行程分割，不同时间段内的行程存储于不同数据块上。时间段划分非常均匀、而且操作简单，更重要的是方便按时间获取行程。对于时间间隔均匀的行程，如浮动车按时间分段存储较好。然而对于时间上分布不均衡的行程，尽管划分的时间间隔相同，但是其数据量可能差别很大。
- 2) 按空间分块存储，即在地理上将数据置于不同的区块内，每个区块内存储经过此地的行程点。由于城市内区域功能不尽相同，均匀的区域划分会导致不同区域数据不均衡。层次划分（四叉树、R 树等）能调节区域内的数据均衡性，是常用的划分方法[16]。在某些情况下，当数据本身就具备一定的区域特性时，直接利用这种特性划分效果更好。例如，手机通信数据中，基站的设置本身就是保证用户通信的均衡性。因此按基站覆盖范围划分数据较为合理。
- 3) 按出行对象分块存储，即每个出行对象的行程单独存储于独立的数据块。由于城市中人群的性质不同，其行程尺度差异较大。对于上班族而言，其出行点基本固定在几个位置；而出租车司机则行程点多且分布在城市各地。简单的按出行对象存储会造成出行尺度小的数据块空间浪费，适当的数据合并是减少空间浪费的主要方式。
- 4) 以上三种方式的混合划分。例如，对于尺度大的行程，先按空间划分再按时间划分，或者先根据出行对象划分，当出行数据集很大时再按照时间划分等。本文的系统中，出行数据主要采用混合划分的方式，确保每个数据块中的出行数据尽量均衡，而且读取性能有所保证。

按照上述规则划分的数据块可以认为是时间或空间上具有关联性的行程集合。因此，可以通过时间、空间值或出行对象等属性唯一性地标识数据块。形式上，数据块是上述三个属性的唯一映射，即：

$$B_i = f(o, t, l)$$

本文实例系统采取的一种存储方式为每个数据块记录一段时间内经过特定地理区域内的所有行程点。数据存储基于 HDFS 文件系统，相邻时段和相邻区域的数据块尽量分布在不同的机器上，使得不相关数据尽量分离。

## 6. 数据索引与检索

有效的索引可以提高行程提取的效率，时间和空间是出行数据两个最重要的维度。利用时空索引将行程关联和组织起来，是高效的分析和可视化的前提。

### 6.1 双向链接哈希索引

可视化查询一般会指定特定的时间和空间范围，范围具有较大的可伸缩性，可以是很大的区域或时间段，也可以是很小的区域或时间段。受查询范围的限制，查询得到的行程往往只是局部。为了提高查询的效率，设计适当的索引是非常必要的。研究人员针对时空数据设计了诸多索引，但索引大都基于较大的时间和空间划分，难以在任意指定的时空范围内快速获取行程路径。

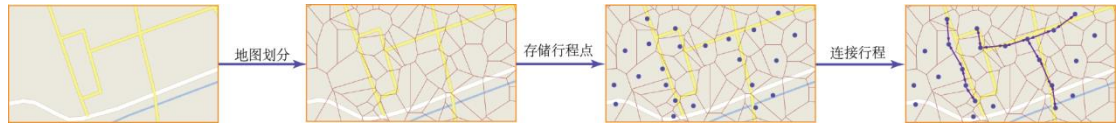


图 2 双向链接哈希索引创建过程

考虑到行程的时间连续性，行程上的点可以依次按照所在空间区域连接。此外，由于同一空间区域内的行程点非常多，为提高查询效率可以在每个空间区域内设立键值对桶，通过哈希函数获取指定出行对象特定时间段的行程。

构建索引的过程（图 2）可分为三个阶段：地图划分、存储行程点和连接行程。在地图划分阶段，根据数据特点将地图分成适当的区域集合，使得经过每个区域的行程点数量尽量平衡，这些区域可以是均匀大小的网格也可以是非均匀的。划分地图后，每个区域都建立一个键值对桶，经过该区域的行程点都将其行程对象及时间戳插入到桶内。最后，相同行程对象按照时间先后顺序双向链接，将各个键值对桶相关串联。

图 3 描述了双向链接的行程索引结构，总体上类似于 Grid-File[19]具有两层索引，不同的是本文的索引要求在数据空间上进行均匀划分，而不是时间或空间上均匀划分。第一级索引定义在时间和空间维度，用于获取对应的键值对桶。第二级索引定义在时间和行程对象维度，用于在键值对桶中唯一地获取行程点数据。

如前所述，本文的数据块可以由时间、空间标识，对应于第一级索引。而数据块内的行程点位置则由第二级索引负责。实现上，键值对桶中的主键是时间戳和行程对象的函数，而值则为数据块内的物理偏移。实际查询中，可以一次读取整个数据块为下次查询提供缓存。

## 6.2 行程查询

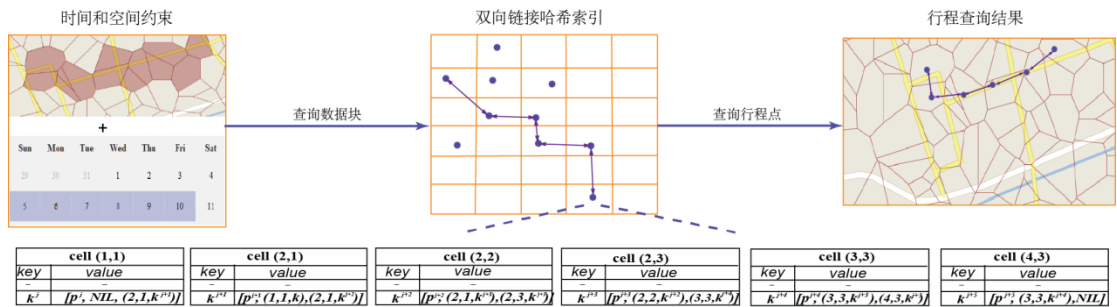


图 3 通过双向链接哈希索引查询行程的过程

行程查询是指定待查询的时间和空间范围，获取满足条件的行程集合。行程查询的一般过程如图 3 所示，可视化查询中分析人员通过图形交互界面选择适当的时间段和地理区域，系统将交互数据转换成查询约束条件，随后在行程索引中得到满足条件的数据索引，最后根据索引在数据块中查找行程点并最终返回行程。

由于不同的分析目标，行程查询可能存在多种模式。单时间段和单地理区域是最简单的，可以分析该时段内经过指定区域的行程变化趋势。多时间段、多地理区域则可用于对比分析。另外，根据行程与选择区域的拓扑关系，还存在四种查询结果：以选择区域为终止点的行程；以选择区域为起点的行程；起点和终点都在选择区域的行程；起点和终点都不在选择区域，只是经过该区域的行程。组合这些查询条件，可以得到 16 种查询模式，如表 1 所示。

表 1 查询模式组合表

	时间（一）	时间（多）	拓扑
空间（一）	X	X	4+4



空间（多）	X	X	4+4
总计	16		

## 7. 出行数据交互界面与可视化

可视化查询需要提供合适的交互工具完成查询条件选取。如前所述，出行数据查询模式主要是对时间和空间进行选取，因而时间和区域选择工具是最基本的。当需要对比多时间段和多区域模式时，选取工具也应能够支持。常用的选取工具如日期列表、日历、时间滑动条、时间仪表盘等可以集成应用。区域选择工具可以用规则化的选择框，也可以使用非规则的套索工具或刷选工具。对于一些特殊的行程，交互界面还应该支持拾取操作可以查看其详细信息。

可视化表达行程是为了更好地理解其特征和模式。行程由多个时序的行程点构成，可以简单地处理为地理空间分布的点集，也可以当作静态轨迹，而动态空间时序图能更好地反映行程特点。不同的处理方法采取的可视编码有很大不同。空间分布点集可以表示成散点图、热力图等。轨迹的可视化方法主要有线图[]、基于图像的方法核密度估计[]、glyph 表示[]等。动态空间时序图可以表示成动态图，但是需要保持空间分布和时序特征。粒子动画可以表示运动对象较少的动态时序图，但是运动对象较多时存在视觉混乱。对于无法在同一视图中表示的数据也可以考虑分开在多个视图中展示不同维度的数据特征。本文采取堆叠图表示行程随时间变化的趋势，而用力引导的边绑定方法展示行程的空间分布。

另外，不同数据源的行程除了时空维度外，还存在其他维度。这些维度可以根据需要分别在不同视图中显示。

## 8. 可视化查询系统实例

本文实现了集成的可视化查询实例系统，包含行程数据组织与存储、行程索引与查询、以及交互界面和行程可视化分析视图。系统根据时间和空间将行程存储于 7 台机器组成的 Hadoop 集群上。为了提高数据查询速度，实例系统根据全局索引读取行程后，数据驻留在内存中。系统基于分布式内存计算模型 Spark 查询行程，保证查询结果能在几百毫秒内返回。

### 8.1 实例系统交互和可视化

实例系统的时间选择采用日历+时间滑动条的方式，其中日历用于选择不同的日期，滑动条可双向拖动用于选择时间段（图 4.a）。区域选择则是将地图分割并刷选网格，不同的数据源地图划分方式不同，如出租车数据采取均匀六边形(图 4.c)而手机通话记录采取 Voronoi 图划分（图 4.b）。

实例系统将行程的时间和空间特征分开在两个视图中表示。行程随时间的变化趋势以堆叠图表示（图 4.b, 4.c 下部），图中每层表示不同速度区间内的出行对象数量。行程在空间上的分布则以静态图表示，为了减少行程中连线的相互影响，应用力引导的边绑定技术勾勒出出行对象在城市内的运动轨迹（图 4.b, 4.c 上部）。

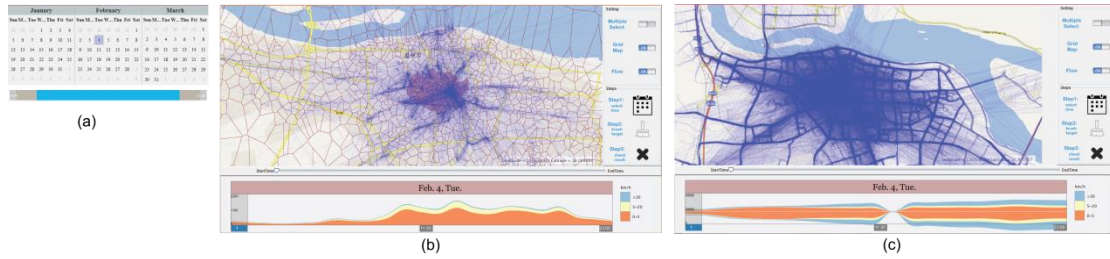


图 4 手机通话记录与出租车 GPS 数据可视化查询结果

## 8.2 实例系统案例分析

实例系统主要针对单一时间和区域的查询,对出租车 GPS 数据和手机通信记录进行可视查询。图 4 显示了对出租车 GPS 数据和手机通话记录进行单日可视化查询之后得到的结果。图 4.b 中的多边形为基站的辐射范围,基站密度大则该区域人口密度大而且属于商业区。图中刷选商业核心区域路段覆盖的基站,经过该路段的人群主要在周边活动。图 4.c 以六边形覆盖道路,当选中商业核心区域街道后查询经过该街道的出租车,可视化结果显示出租车在更广泛的范围内活动,但是在选中区域内活动更频繁。

从两种数据时间活动情况来看,手机用户从上午八点后直到夜间十一点人群活动比较频繁。特别是上午 11 点以及下午 1 点左右出现人群活动峰值,但是出租车在中午 12 点左右很少活动(此时段为出租车司机交接班)。由此可以看出数据往往存在一定的偏差,如果仅从出租车数据来分析城市居民出行模式可能得出错误结论,而综合这两类数据则能更好地解释出行模式。

## 9. 结论

城市出行数据对研究城市复杂问题具有重要意义,然而已有的方法大都针对单数据源或单个问题。可视化查询能够面向多数据源而且分析目标可以迭代产生。本文提出了面向多源出行数据的可视化查询模型。查询模型从数据抽象、数据组织和管理、查询交互界面以及可视化分析等方面进行探讨,介绍了一系列方法和技术。最后实现了查询模型的实例系统,从两类出行数据查询的分析结果看,可视化查询模型不仅能发现城市居民出行规律而且避免了单数据源分析导致的理解偏差。

致谢: 本文特别感谢浙江大学 CAD&CG 国家重点实验室郑文庭老师及组内同学的长期帮助和支持。

## 参考文献

- [1] Yu Zheng, Yanchi Liu, Jing Yuan, Xing Xie, Urban computing with taxicabs, UbiComp 2011
- [2] Visual Traffic Jam Analysis Based on Trajectory Data, *IEEE Transactions on Visualization and Computer Graphics (VAST'13)*, 19(12):2159-2168, 2013.
- [3] Yu Zheng, Furui Liu, Hsun-Ping Hsie. U-Air: when urban air quality inference meets big data. KDD 2013
- [4] X. Song, Q. Zhang, Y. Sekimoto, T. Horanont, S. Ueyama, R. Shibasaki. Modeling and probabilistic reasoning of population evacuation during large-scale disaster. KDD 2013
- [5] Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-László Barabási. Limits of Predictability in Human Mobility, *Science* 19 February 2010: 327 (5968), 1018-1021
- [6] Marta C. González, Cesar A. Hidalgo R., and Albert-László Barabási. Understanding individual human mobility

patterns

- [7]Desheng Zhang, Exploring human mobility with multi-source data at extremely large metropolitan scales.
- [8] Gonzalez, M. C., Hidalgo, C. A. & Barabasi, A.-L. Understanding individual human mobility patterns. *Nature* 453, 779–782 (2008).
- [9]Song, C., Qu, Z., Blumm, N. & Barabási, A.-L. Limits of predictability in human mobility. *Science* 327, 1018–1021 (2010).
- [10]Yu Zheng, Xiuwen Yi, Ming Li, Ruiyuan Li, Zhangqing Shan, Eric Chang, Tianrui Li. Forecasting Fine-Grained Air Quality Based on Big Data. In the Proceeding of the 21th SIGKDD conference on Knowledge Discovery and Data Mining
- [11] Nivan Ferreira , Jorge Poco , Huy T. Vo , Juliana Freire , Cláudio T. Silva, Visual Exploration of Big Spatio-Temporal Urban Data: A Study of New York City Taxi Trips, *IEEE Transactions on Visualization and Computer Graphics*, v.19 n.12, p.2149-2158, December 2013
- [12] Y. Theodoridis, M. Vazirgiannis, and T. Sellis. Spatio-temporal indexing for large multimedia applications. In *Proceedings of the IEEE International Conference on Multimedia Computing and Systems*, pages 441– 448. IEEE, 1996.
- [13] K.Z. K.Deng ,K.Xie and X.Zhou. Trajectory indexing and retrieval. In *Computing with Spatial Trajectories*, pages 35–60. Springer, 2011
- [14] M. A. Nascimento and J. R. Silva. Towards historical r-trees. In *Proceedings of the ACM symposium on Applied Computing*, pages 235–240. ACM, 1998.
- [15] Lauro Lins, James T. Klosowski, and Carlos Scheidegger. Nanocubes for Real-Time Exploration of Spatiotemporal Datasets. *Visualization and Computer Graphics*, *IEEE Transactions on* 19, no. 12 (2013): 2456-2465
- [16] V. Prasad Chakka and Adam Everspaugh and Jignesh M. Patel, Indexing Large Trajectory Data Sets With SETI, *CIDR* 2003
- [17] Philippe Cudré-Mauroux, Eugene Wu, and Samuel Madden. TrajStore: An Adaptive Storage System for Very Large Trajectory Data Sets. *26th International Conference on Data Engineering (ICDE)* 2010).
- [18] Shneiderman, Ben, Dynamic Queries for Visual Information Seeking, in *IEEE Software*, 11(6): 70-77, 1994.
- [19] J. Nievergelt, H. Hinterberger The Grid File: An Adaptable, Symmetric Multikey File Structure. *Institut für Informatik, ETH and K. C. Sevcik*, 1984. Abstract, pp.1.